

Application No. 09/872,581
RCE to OA of 06/09/2006

Remarks

In the present response, two claims (1 and 12) are amended. Claims 1-2 and 4-26 are presented for examination. No new matter is presented.

I. Examiner Interview

Applicant (Fabio Casati) and patent attorney (Philip S. Lyren) thank Examiner Lin for having a telephone interview on August 7, 2006. During this interview, the Examiner agreed to withdraw all current section 112 rejections if claims 1 and 12 were amended as follows:

Claim 1

- b) ~~during~~at run time determining a number of work nodes to be activated in the multinode ~~equal to~~-based on a number of elements in a vector;
- c) ~~activating~~executing the number of work nodes in the multinode;

Claim 12

determining a number of multiple instances of same work nodes in one of the multinodes to be activated ~~equal to~~-based on a number of elements in a vector;
~~activating~~executing the number of work nodes in the one of the multinodes....

Claims 1 and 12 are amended as agreed in the interview.

II. Request for Webopedia References

In the Appendix, Applicants provide definitions copied from www.webopedia.com for the terms "variable" and "vector" as requested by the Examiner.

III. Claim Rejections: 35 USC § 112

Claims 1-2 and 4-20 are rejected under 35 USC § 112, first paragraph, as failing to comply with the enablement requirement. In light of the interview and amendments to

Application No. 09/872,581
RCE to OA of 06/09/2006

claims 1 and 12, this rejection is moot. Applicants respectfully request withdrawal of this rejection.

IV. Claim Rejections: 35 USC § 112

Claims 1-2 and 4-20 are rejected under 35 USC § 112, first paragraph, as failing to comply with the written description requirement. In light of the interview and amendments to claims 1 and 12, this rejection is moot. Applicants respectfully request withdrawal of this rejection.

V. Claim Rejections: 35 USC § 112

Claims 1-2 and 4-20 are rejected under 35 USC § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter of the invention. In light of the interview and amendments to claims 1 and 12, this rejection is moot. Applicants respectfully request withdrawal of this rejection.

VI. Claim Rejections: 35 USC § 103

Claim 21 is rejected under 35 USC § 103 as being unpatentable over USPN 6,041,306 (hereinafter Du) in view of Applicant Admitted Prior Art (AAPA). This rejection is traversed.

To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art cited must teach or suggest all the claim limitations. *See* M.P.E.P. § 2143. For at least the following reasons, Applicants assert that the rejection does not satisfy these criteria.

Claim 21 recites numerous recitations that are not taught or suggested in Du in view of AAPA. By way of example, claim 21 recites “determining, based on an activation rule, whether the activation of the multiple parallel instances of the same work node is a resource-based activation or a variable-based activation” (emphasis added).

Application No. 09/872,581
RCE to OA of 06/09/2006

The Office Action cites Du at column 2, lines 22-51 and column 6, lines 1-6. Applicants respectfully disagree.

Column 2, lines 22-51 in Du discusses a resource manager in a WFPM and a need for a flexible WFPM that dynamically redefines the relationship with the resource managers. Notice, nowhere does this section of Du teach or suggest that multiple instances of the same work node are activated based on either resource-based activation or variable-based activation. **Du is completely silent on variable-based activation.**

Column 6, lines 1-6 in Du teaches a workflow management interface that allocates resources to a task according to "availability of the resources using the workflow management modules." Notice, nowhere does this section of Du teach or suggest that multiple instances of the same work node are activated based on either resource-based activation or variable-based activation. **Du is completely silent on variable-based activation.**

Further, AAPA states that "the number of activities that are executed in parallel is **always equal** to the number of resources that are available for execution of that activity" (emphasis added: p. 5, lines 10-13). Notice, nowhere does AAPA teach or suggest that multiple instances of the same work node are activated based on either resource-based activation or variable-based activation. **AAPA is completely silent on variable-based activation.** Further, AAPA expressly teaches away from variable-based activation since AAPA states that the number of activities being executed is **always equal** to available resources.

For at least these reasons, claim 21 and its dependent claims are allowable over Du in view of AAPA.

VII. Claim Rejections: 35 USC § 103

Claims 22-24 are rejected under 35 USC § 103 as being unpatentable over USPN 6,041,306 (hereinafter Du) in view of Applicant Admitted Prior Art (AAPA) and US 2002/0083166 (hereinafter Dugan). Applicants respectfully traverse.

As noted above in section VI, Du and AAPA do not teach or suggest all the limitations of independent claim 21. Dugan does not cure the deficiencies of Du and

Application No. 09/872,581
RCE to OA of 06/09/2006

AAPA. Thus, for at least the reasons given above in connection with independent claim 21, dependent claims 22-24 are allowable over Du and AAPA in view of Dugan.

VII. Claim Rejections: 35 USC § 103

Claims 25-26 are rejected under 35 USC § 103 as being unpatentable over USPN 6,041,306 (hereinafter Du) in view of Applicant Admitted Prior Art (AAPA) and US 2002/0083166 (hereinafter Dugan). Applicants respectfully traverse.

Claim 25 recites:

reading an activation rule to determine if activation of the
multinode is based on **resource-based activation or variable-
based activation.**

The Examiner argues that these recitations are shown in column 2, lines 22-51 and column 6, lines 1-6 of Du. Applicants respectfully disagree.

Column 2, lines 22-51 in Du discusses a resource manager in a WFPM and a need for a flexible WFPM that dynamically redefines the relationship with the resource managers. Notice, nowhere does this section of Du teach or suggest that multiple instances of the same work node are activated based on either resource-based activation or variable-based activation. **Du is completely silent on variable-based activation.**

Column 6, lines 1-6 in Du teaches a workflow management interface that allocates resources to a task according to "availability of the resources using the workflow management modules." Notice, nowhere does this section of Du teach or suggest that multiple instances of the same work node are activated based on either resource-based activation or variable-based activation. **Du is completely silent on variable-based activation.**

If the Examiner maintains this rejection, then Applicants respectfully ask the Examiner to identify a citation in Du that teaches or suggests "variable-based activation."

For at least these reasons, claim 25 and its dependent claims are allowable over the art of record.

Application No. 09/872,581
RCE to OA of 06/09/2006

As another example, claim 25 recites activating the multinode to execute a number of plural instances of the same work node. The claim then recites that **the number is read from a variable name that is a vector when the activation rule is the variable-based activation**. These recitations are not taught in the art of record.

The Office Action admits that Du and AAPA do not teach these recitations. Instead, the Office Action argues that these recitations are taught in Dugan at paragraphs [0067 – 0069]. These sections in Dugan discuss responsibilities of a Service Administration according to its FIG. 5. For example, the Service Administration component activates data and service components and configures service nodes in accordance with information the SA receives. Nowhere does Dugan teach or suggest that the Service Administration component activates a multinode to execute a number of plural instances of the same work node, wherein the “the number is read from a variable name that is a vector when the activation rule is the variable-based activation.” For example, in which paragraph does Dugan suggest activating a given number of plural instances of the same work node based on a variable name that is a vector? As another example, in which paragraph does Dugan suggest an activation rule that is “variable-based activation?” If the Examiner maintains this rejection, Applicants respectfully ask for clarification on which sections of Dugan teach such recitations.

For at least these reasons, claim 25 and its dependent claims are allowable over the art of record.

Application No. 09/872,581
RCE to OA of 06/09/2006

CONCLUSION

In view of the above, Applicants believe all pending claims are in condition for allowance. Allowance of these claims is respectfully requested.

Any inquiry regarding this Amendment and Response should be directed to Philip S. Lyren at Telephone No. 832-236-5529. In addition, all correspondence should continue to be directed to the following address:

Hewlett-Packard Company
Intellectual Property Administration
P.O. Box 272400
Fort Collins, Colorado 80527-2400

Respectfully submitted,

/Philip S. Lyren #40,709/

Philip S. Lyren
Reg. No. 40,709
Ph: 832-236-5529

Application No. 09/872,581
RCE to OA of 06/09/2006

Appendix

Definition for Variable

A symbol or name that stands for a value. For example, in the expression

$x+y$

x and y are variables. Variables can represent numeric values, characters, character strings, or memory addresses.

Variables play an important role in computer programming because they enable programmers to write flexible programs. Rather than entering data directly into a program, a programmer can use variables to represent the data. Then, when the program is executed, the variables are replaced with real data. This makes it possible for the same program to process different sets of data.

Every variable has a name, called the *variable name*, and a data type. A variable's data type indicates what sort of value the variable represents, such as whether it is an integer, a floating-point number, or a character.

The opposite of a *variable* is a constant. Constants are values that never change. Because of their inflexibility, constants are used less often than variables in programming.

Definition for Vector

(1) In computer programming, a one-dimensional array. A vector can also mean a pointer.

(2) In computer graphics, a line that is defined by its start and end point.